

Code Churn

1) Какво представлява Code Churn?

Code Churn е една естествена част от процеса на разработката на един софтуер. Тази метрика се изразява в това колко често даден код бива пренаписван или изтриван много скоро след като е бил написан.

Разбира се, Code Churn зависи от проект на проект, тоест от това какъв тип проблем се решава (нов или познат), мотивацията на разработчиците и степента на тяхното разбиране на проблема, както и от недвусмислието на изискванията от страна на заинтересованите лица.

Например, в последните две седмици в даден проект биват написани 100 нови реда код, но в същото време биват премахнати или променени 20 реда от тях. По този начин, имаме 20% Code Churn.

Всеки мениджър на даден проект трябва да знае в даден момент какъв процент Code Churn е “нормален” за проекта, чрез който да следи дали има проблеми в разработката, екипа, изискванията и т.н.

2) Защо ни е Code Churn?

Code Churn е метрика, която помага до голяма степен на мениджърите на проектите, когато тя бъде използвана по правилен начин. Чрез следенето на Code Churn всекидневното и определянето на отклонението от нормалния процент, един мениджър може да открие проблеми в разработката отрано. Такива проблеми, особено точно преди доставянето на софтуера, могат да бъдат фатални и затова тяхното премахване е от голямо значение. Ето защо Code Churn е една метрика, използването на която може да се окаже от голямо значение за успеха на разработката на даден софтуер.

3) Аргументи “За”:

Според мен, Code Churn е доста силен “инструмент”, чрез който могат да се отстраняват проблеми, които застрашават успеха на разработката на даден софтуер. Тази метрика ни “показва” кога даден проблем може да се появи, сигнализирайки ни по този начин да съставим такива тактики, чрез които да го избегнем или поправим, без да се засяга твърде много разработката. Така, ние можем да гарантираме (поне донякъде) нормалното протичане на процеса на разработка на даден софтуер.

4) Аргументи “Против”:

Code Churn може да бъде “мощно средство” за откриване и премахване на проблеми, но в същото време тази метрика може да бъде много

трудна за употреба. Какво означава “нормален процент” на Code Churn в рамките на

даден интервал от време? Това е най-важният въпрос, на който един мениджър трябва да си отговори, преди да използва тази метрика. Правилното определяне на този процент ще доведе до правилното определяне на проблемите. Но ако не е, в такъв случай могат да се появят допълнителни проблеми, които да застрашат успеха на разработката. В много случаи този процент не е толкова лесен за определяне. Ето защо, Code Churn е единствено полезен тогава и само тогава, когато бъде използван и определен правилно.

5) Заключение:

Code Churn е една метрика, която със сигурност би допринесла значително за справянето с проблемите, които се появяват по време на разработката на даден софтуер. Но тя трябва да се използва внимателно, тъй като не винаги може да служи за правилен ориентир.

6) Източници:

1. Introduction to Code Churn - November 13, 2019

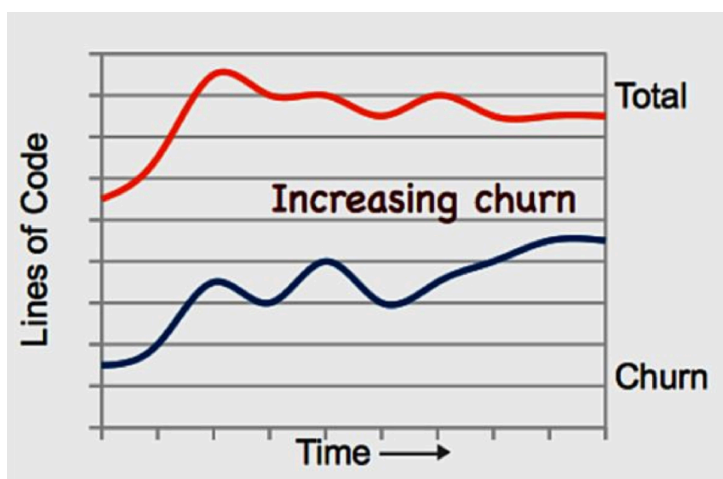
- <https://www.pluralsight.com/blog/tutorials/code-churn>

2. What is Code Churn and How to Reduce it? - September 10, 2019

- <https://textexpander.com/blog/what-is-code-churn-and-how-to-reduce-it/>

Коментар:

Високата степен на code churn не е проблем сам по себе си. Това е по-скоро симптом, защото кодът се променя по някаква причина. Може би имаме област/свойство с характеристики, което е слабо разбрано. Или може би просто имаме модул с нискокачествена реализация. Като се има предвид причините, едва ли е изненадващо, че code churn е добра прогноза за дефекти.



Ето един Churn сценарии, в който би било страшно да се озовете. Както се вижда от графиката по-горе, приближавайки deadline-а на проекта, той продължава прогресивно даменя повече редове от кода си. Тъй като има положителна корелация между code churn и дефектите, този патърн означава, че слагаме качеството на нашия код под риск.

Този churn патърн означава, че проекта няма да достигне своя deadline. Има още много работа преди да достигнем нещо наподобяващо стабилна база код. Тази метрика може да издаде и какъв тип растеж имате. В случая - не добър.

Определено това е една доста интуитивна метрика за измерване на мащабируемостта (scalability-то). Освен за това може индиректно да служи за индикатор доколко разработчика е мислел за мащабируемост докато е писал кода. Тоест този code churn индекс индиректно оценява и уменията на разработчика. Ако пък прекалено голямо количество код се премахва при несъществена промяна това дори може да опира и до некачествено архитектурно решение.